DIME Dynamic Documentation Training
Stata Exercise

Luiza Andrade & Mrijan Rimal

May 24, 2017

Introduction

This exercise introduces you to how to export files from Stata that can be read in LaTeX. See exercise 1 and 2 for instructions on how to import file into LaTeX. After this exercise and exercise 1, you will have a document that is automatically updated each time you run your Stata and your LaTeX code.

We have provided you with a do-file that has code that creates file that you can import in a LaTeX document. We will go through these examples and then we will ask you to create some tables and graphs of your own using your own data. Note that this is not an exercise on Stata, only on exporting in LaTeX format from Stata, so this exercise assumes knowledge of some intermediate level Stata commands.

Part 1: Setting up a Folder Structure

Do not rush over this part. You will be forced to write an necessarily complicated LaTeX code unless you set up a simple but well organized folder structure for where to export tables and figures from Stata. We strongly recommend that you start by setting up the following folder structure.

Create one folder called `Output`, and inside this folder, create a folder called `Raw`. In the `Raw` folder you will export tables and graphs from Stata that you will import in your LaTeX document. Your LaTeX document will be saved in the `Output` folder.

See the example below. As your project grows bigger it is common that sub-folders are added in the `Raw` folder. For example, tables and figures often have separate folders. You will find your preferred way to organize this.

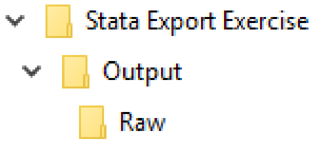


Figure 1: Common Folder Structure

Part 2: Setting your path for Stata

We now need to tell all commands in Stata to save the tables and figures they export to the `Raw` folder. We want to create a global with the file path pointing to the `Raw` folder that we can use in our commands. Using a global instead of typing out the folder location in all commands both makes the code simpler, and makes it easier to update if you would have to move your folders.

Open the do-file `Export tables and images.do` that you find in the same folder as this handout. Look for the part that says:

```
global main_folder "<<<ENTER YOUR FOLDER PATH HERE>>>"
```

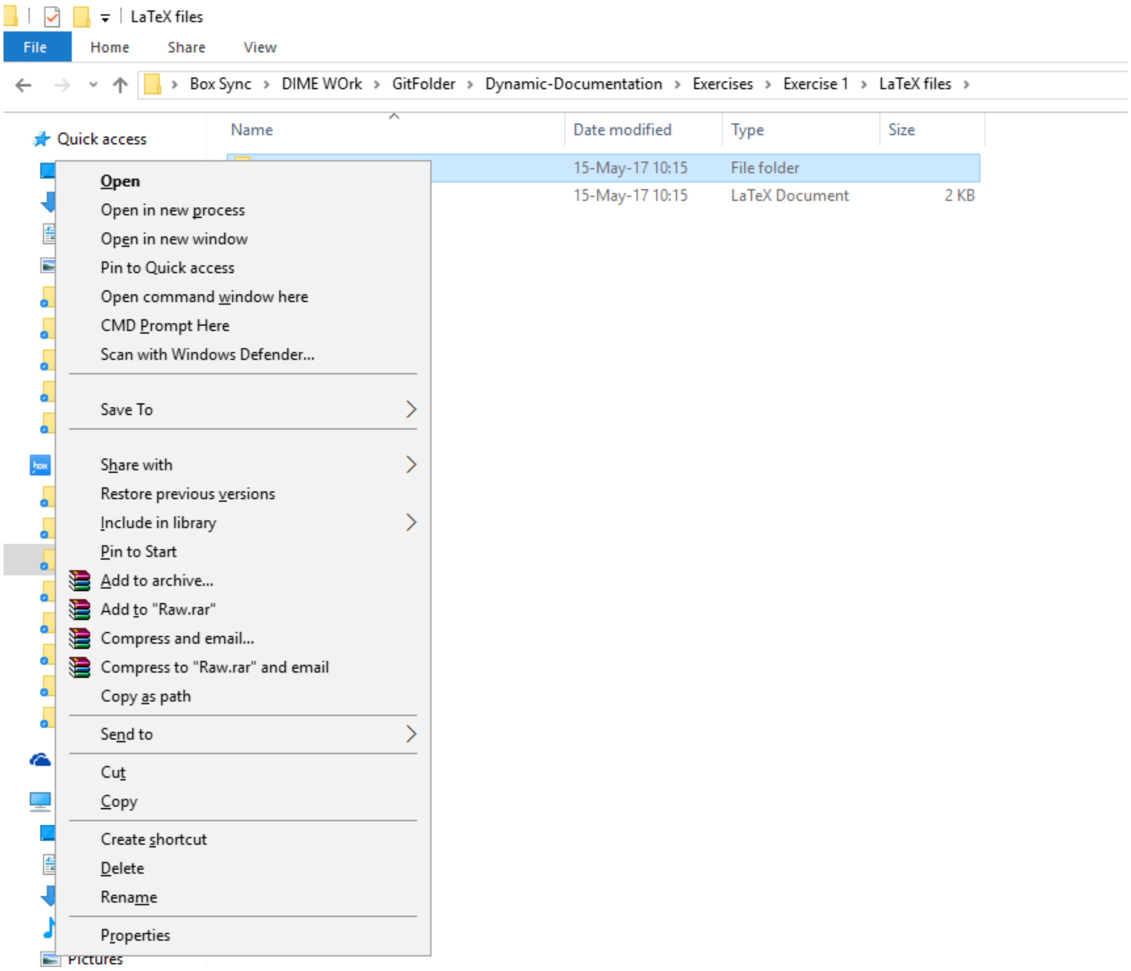
You need to enter the path to the folder you created in part 1. If you have a Windows computer and you created your `Output` folder in your Documents folder, then your file path should look similar to this:

```
global main_folder "C:\Users\JoeSmith\Documents\Output"
```

The following two sub-sections help you find the file path to the folder you just created if you are not sure how to do that. The first sub-section gives advice for Windows, and the second for Mac.

Finding Path on a Windows Computer

Path to a file can be found by selecting a file and pressing SHIFT on your keyboard and RIGHT CLICKING your mouse and then clicking COPY AS PATH in the resulting drop down menu as shown in Figure 2.



Name	Date modified	Type	Size
	15-May-17 10:15	File folder	
	15-May-17 10:15	LaTeX Document	2 KB

Figure 2: Finding path on a Windows Computer - Solution 1

Another solution to finding the path on a Windows computer is shown in Annex 1.

Finding Path on a Mac

Part 3: File Formats and Names

Not all file formats that you can chose when exporting from Stata is valid to import into LaTeX. Figures must be saved in `.png` format. All figures produced in Stata can be exported in `.png`. Even if some commands are not able to export in this format, you can use Stata to convert the figure for you. You will find more details on this in the parts below where we export figures.

Tables must be saved in `.tex` format. Tables cannot be converted as easily as figures, so it is usually easier to find an alternative command that produces the same table but can export to `.tex` format. Most commands that export tables are able to export to this format so it should not be difficult to find an alternative.

Files exported in for example `MS Word` or `MS Excel` or in Stata's `.gph` format are not valid to import to LaTeX.

Files exported from Stata should have a descriptive name. Otherwise you increase the risk for human errors, and the whole point of dynamic documents is to reduce exactly that. For example - rather than `graph1.png`, `graph2.tex`, names like `treatmentEffectGraph.png` would make it easier to understand what files we are using.

Part 4, Task 1: Tabulate Categorical Variables

The code for this Part 4, 5 and 6, are all in the do file `Export tables and images.do`. The data that you will use is a sample data set that all instances of Stata alrady have. So you do not need a data file, the do-file will load this data set for you.

First we want to start with exporting a simple tabulate table of frequencies. We first use the command `tabulate` to generate the statistics, and use `esttab` to export it to `.tex` format. The command `estpost` that comes before the `tabulate` command takes the result for the tabulation and prepare them in a format that `esttab` can use. See the help files for `estpost` and `esttab` for more details on how this works. This is an exercise in LaTeX and not the commands in the `estout` package so we will not go into more details about this at this point.

To export the tabulation in `.tex` format using `esttab`, you simply set the file extension in the `using` option. Like this: `using "$raw_output/categorical.tex", replace`. Remember to always have an explanatory file name.

Now check your `Raw` folder to see the `categorical.tex` file you have exported.

Part 4, Task 2: Regression table

Similar to the previous task, we can use `esttab` to export regression tables. First we run the regressions we want to export the results for. The command `eststo:` before the regression commands stores the result of the regressions so that `esttab` can export all of them together. You can use any option to the regression that Stata allows. If Stata can run the regression and display the regression results in the Stata window, then `esttab` will be able to export those results to `.tex` format.

The difference between the two regressions is that the second regression includes fixed effects for region. We can use the command `estadd` to add text to the tables. In this example we use this to indicate which regression that included fixed effects. See the help files for `estout`, `estadd` and `esttab` for more details on how this works. This is an exercise in LaTeX and not the commands in the `estout` package so we will not go into more details about this at this point.

Similarly to the previous task, export the stored regression results using `esttab` to `.tex` format bys setting the file extenstion in the `using` option. Like this: `using "$raw_output/regression_table.tex"`. `esttab` will export all results stored in your estimation memory. That is why it is important that we start each task with the code `estimates clear` in the do file. Otherwise we might add results from previous tables to this table.

Now check your `Raw` folder to see the `regression_table.tex` file you have exported.

Part 5, Task 1: Manually Create a Graph and then Export it

This task shows us how to export graphs created in Stata to export in a format that LaTeX can read. Stata's save graph feature saves the graph in `.gph` format which only Stata can read. However, the `graph export` feature saves the graph in a picture format i.e. png format which can be read by the photo viewer on your computer, phone and also LaTeX. It is *absolutely critical* to export the graph in this format, so that LaTeX can import it.

Using the `graph export "$raw_output/regular_graph.png", width(5000) replace` exports the graph in png format which LaTeX can read.

Note: This is different from the `iegraph` command where save can both export the graph in `.gph` and `.png` format. For Stata's `graph` command, export has be to be used to make it readable in LaTeX.

Now check your `Raw` folder to see the `regular_graph.png` file you have exported.

Part 5, Task 2: Using `iegraph` to create a figure

While Stata's `graph twoway` uses the `save` feature to export pictures in a `.gph` format, and we have to use the `graph export` to export it in png format. Many commands, for example - `iegraph`, can export directly to either format using the same `save` option.

This exercise teaches how to use `iegraph` to create a figure and export it to the graphs folder. On `iegraph`, while exporting a picture for LaTeX, always make sure that the picture has the extension `.png` at the end of the filename. **Without that, `iegraph` is just going to export the picture in a format which only Stata can read!**

Using the `save("$raw_output/iegraph.png")` ensures that the graphs are directly saved to the specified output folder.

Now check your `Raw` folder to see the `iegraph.png` file you have exported.

Part 6: Making a Dynamic Document

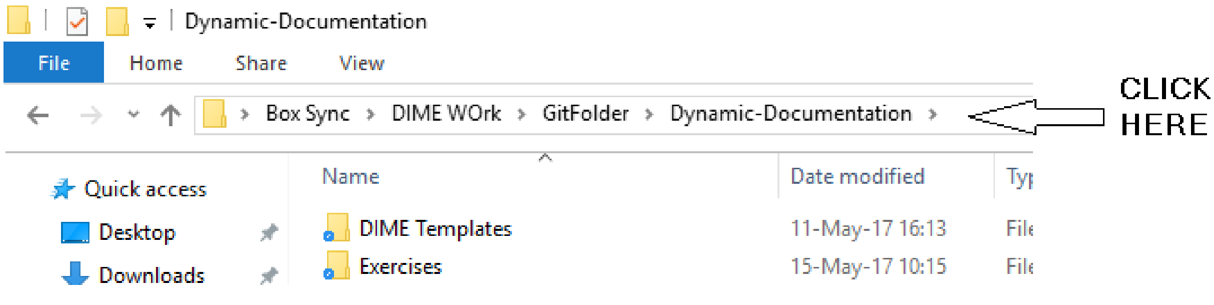
Here, we will produce a dynamic document. Please only do this do **if** you have completed up to `Part 5, Task 2` of the exercise.

1. Save the pdf created upto now in a separate location from your `.tex` file.
2. Go to `line 60` of the Stata do file and change the seed from `215320` to a number different than `215320`.
   Note: Changing seeds is not recommended for actual projects. The change in seeds here is used to highlight the change in treatment/control groups.
3. Rerun the do-file with the new seed.
4. Now, open the earlier created LaTeX file and press *Build and Compile* under the *Tools*.
5. Now if you compare the pdf file you have just generated with the one you saved earlier, you will find that the tables would have updated automatically.

Part 7, Using a do-file to edit a .tex file after exporting it

During this part of the exercise, you will learn how to use commands in Stata to format your tables. While tables exported from Stata to LaTeX are generally very nice, sometimes they need to be tweaked a little to make them look nicer. So, in this exercise, you'll use the `filefilter` command in Stata to make small changes to the files exported by Stata.

Annex 1

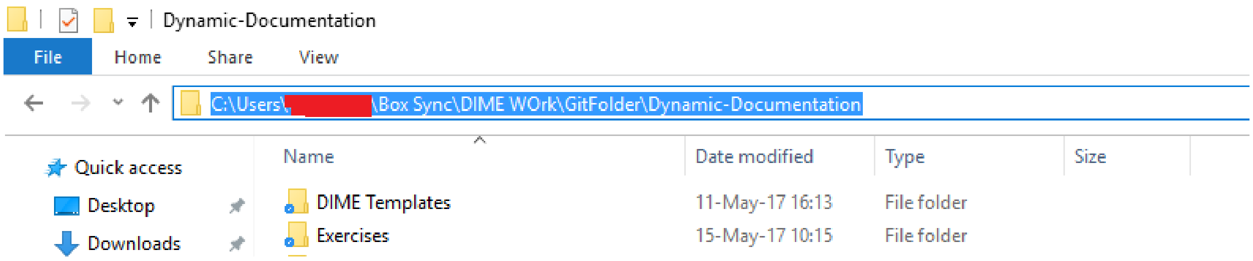


CLICK HERE

Name	Date modified	Typ
DIME Templates	11-May-17 16:13	File
Exercises	15-May-17 10:15	File

Figure 3: Finding Path on a Windows Computer

As shown in Figure 3, left clicking(normal click) on the bar at the top of the `File Explorer` windows where our files are saved shows us the complete path to the files in a Windows computer.



C:\Users\[redacted]\Box Sync\DIME WOrk\GitFolder\Dynamic-Documentation

Name	Date modified	Type	Size
DIME Templates	11-May-17 16:13	File folder	
Exercises	15-May-17 10:15	File folder	

Figure 4: Path shown on a Windows Computer

We can see in Figure 4, that the complete path to the folder is shown. We can then paste this path when setting the path in our Stata do-file and changing the path where it says

```
global main_folder ‘‘<<<ENTER YOUR FOLDER PATH HERE>>>’’
```